UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/646,456	08/21/2003	Richard Clark	019680-006600US	9524

20350 7590 11/03/2006

TOWNSEND AND TOWNSEND AND CREW, LLP
TWO EMBARCADERO CENTER
EIGHTH FLOOR
SAN FRANCISCO, CA 94111-3834

EXAMINER
PANTOLIANO JR, RICHARD

ART UNIT	PAPER NUMBER
2194	

DATE MAILED: 11/03/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	**Application No.** 10/646,456	**Applicant(s)** CLARK ET AL.	
	Examiner Richard Pantoliano Jr	**Art Unit** 2194	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *25 August 2003*.

2a)□ This action is **FINAL**. 2b)☒ This action is non-final.

3)□ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-36* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)□ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-36* is/are rejected.

7)□ Claim(s) _____ is/are objected to.

8)□ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)□ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *25 August 2003* is/are: a)☒ accepted or b)□ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)□ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)□ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)□ All b)□ Some * c)□ None of:

1.□ Certified copies of the priority documents have been received.

2.□ Certified copies of the priority documents have been received in Application No. _____.

3.□ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)

2) □ Notice of Draftsperson's Patent Drawing Review (PTO-948)

3) ☒ Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date *20041227*.

4) □ Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ .

5) □ Notice of Informal Patent Application

6) □ Other: _____.

DETAILED ACTION

1. This is the initial office action for Application# **10/646,456** filed on **23 August 2003**. **Claims 1-36** are currently pending and have been considered below.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

> Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.

3. **Claims 27-36** are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Claims 27-36 are not limited to tangible embodiments. In view of Applicant's disclosure, specification page 12, paragraph [0042], the medium is not limited to tangible embodiments, instead being defined as including both tangible embodiments (e.g., CDs and DVDs) and intangible embodiments (e.g., data signals embodied in a carrier wave). As such, the claim is not limited to statutory subject matter and is therefore nonstatutory.

4. To overcome this type of 101 rejection the claims need to be amended to include only the physical computer media and not a transmission media or other intangible or non-functional media.

Claim Rejections - 35 USC § 103

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

6. **Claims 1-36** are rejected under 35 U.S.C. 103(a) as being unpatentable over De Armas et al (US Pat: 6,611,878) in view of Teilhet (Teilhet, Stephen. Subclassing and Hooking with Visual Basic. O'Reily Publishing, June 2001)

7. **Claim 1:** De Armas et al discloses the invention substantially as claimed including a method for use by a first process executing in a computer system for interacting with a second process executing in the computer system, the method comprising:

a) during a startup sequence of the second process, creating a copy of a global notification hook of the first process in the second process *(Col 7, Lines 16-63)(The installation of the global hook causes it to be installed upon startup of a new application)*;

b) using the copy of the global notification hook, detecting an occurrence of a triggering message passed between an operating system and a thread of the second process *(Col 8, Lines 5-34)(The "PostMessage()" API call is used to send the triggering message and the "GetMessage()" API call is used to retrieve said message)*; and

c) in response to detecting the occurrence of the triggering message, determining whether subsequent messages passed between the operating system and the thread of the second process should be monitored *(Col 8, Lines 5-34).*

8. While De Armas et al discloses the interception of further messages and modifying the behavior of the window to which those messages are directed *(De Armas et al; Col 7, Lines 56-62)*, De Armas et al does not disclose the use of a thread-level hook to intercept and modify said messages.

9. Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept messages directed to a window to modify the functionality of said window *(Teilhet; Chpt 18.2, Pgs 3-5)*.

10. It would have been obvious to one of ordinary skill in the art at the time of invention to modify the method disclosed by De Armas et al with Teilhet's teachings concerning the WH_CBT thread-level hook. One of ordinary skill would have been motivated by the fact that De Armas et al discloses the general use of thread-level hooks *(De Armas et al; Col 7, Lines 5-15)* and that Teilhet discloses that thread-level hooks can be used as a substitute for the functionality of the method of message interception disclosed by De Armas et al, subclassing *(Teilhet; Chpt 3.2, Pg 1, 4th paragraph)*.

11. **Claim 2:** De Armas et al discloses causing an action to occur in response to intercepted messages *(Cols 11 and 12)* and Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept messages directed to a window to modify the functionality of said window *(Chpt 18.2, Pgs 3-5)*.

12. **Claim 3:** De Armas et al further discloses the action including creating a visual effect for a window of the second process *(Col 11, Line 65 - Col 12, Line 11)*.

13. **Claim 4:** De Armas et al further discloses the communication occurring within the second process not affecting the operation of a third process that is concurrently operating *(Col 9, Lines 36-60) (Use of the "SendMessage()" API to send messages directly to the recipients of said message).*

14. **Claim 5:** Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept a window creation message directed to a window to trigger an event *(Teilhet; Chpt 18.2, Pg 5).*

15. **Claim 6:** De Armas et al further discloses determining a window of interest when deciding to monitor further messages *(Col 8, Lines 10-42) (A window of interest is one that currently has system focus or input has been received from the user to indicate a window of interest).*

16. **Claim 7:** This claim is rejected for the same reasons as **Claim 6** above.

17. **Claim 8:** De Armas et al further discloses the first process is a desktop management process *(Col 5, Lines 26-49) (The TIS system intercepts messages to all* *desktop applications and either modifies messages being transmitted or forwards them, to the necessary application).*

18. **Claim 9:** De Armas et al further discloses the second process is an application process *(Col 5, Lines 26-49).*

19. **Claim 10:** De Armas et al further discloses said method further comprising, during a startup of the first process:

a) detecting a third process executing in the computer system *(Col 7, Lines 16-63) (The detecting of all processes meets this limitation)*;

b) inserting a copy of the global notification hook into the third process *(Col 7, Lines 16-63) (The injection of the DLL in all processes meets this limitation)*; and

c) broadcasting a private startup message to the copy of the global notification hook in the third process *(Col 7, Line 64 - Col 8, Line 34).*

20. **Claim 11:** De Armas et al further discloses wherein, in response to the private startup message, the copy of the global notification hook executes act of determining whether subsequent messages passed between the operating system and a thread of the third process should be monitored *(Col 7, Line 64 - Col 8, Line 42).*

21. **Claim 12:** De Armas et al further discloses determining whether subsequent messages passed between the operating system and the thread of the third process should be monitored includes:

a) identifying a previously created window of the third process *(Col 8, Lines 10-34) (A window can be identified by requesting the currently in-focus window or can be a user -selected window)*; and

b) determining whether the previously created window is of interest, wherein subsequent messages should be monitored in the event that the previously created window is of interest *(Col 8, Lines 10-34) (A window of interest is one that currently has system focus or indicated as such by a user).*

22. **Claim 13:** De Armas et al further discloses the act of mapping executable code into an address space of the second process for the code that will intercept further messages *(Col 7, Lines 16 - Col 8, Line 9 and Col 9, Lines 18 - 35).*

23. **Claim 14:** De Armas et al further discloses the act of creating the copy of the global notification hook includes mapping executable code for the global notification hook into an address space of the second process *(Col 7, Lines 16 - Col 8, Line 9).*

24. **Claim 15:** De Armas et al further discloses the act of detecting the occurrence of the triggering message includes receiving message data of the triggering message *(Col*

8, Lines 26-34) (The flag in the form of a message containing a unique number meets this limitation).

25. Claim 16: De Armas et al further discloses the message data of the triggering message is provided to the copy of the global notification hook concurrently with a transmission of the triggering message to the thread of the second process *(Col 7, Lines 5 - Col 8, Line 34) (Since the global notification hook is installed within the second process and runs within the context of the second process's thread, the delivery of the message to the global notification hook meets the the further limitations of this claim).*

26. Claim 17: De Armas et al further discloses the second process has a process-specific message queue that receives the transmitted message data of the triggering message *(Col 4, Lines 37 - 63 and Col 7, Lines 16-55) (The "GetMessage()" function removes messages from the disclosed message queue of the second process, thereby meeting this further limitation).*

27. Claim 18: De Armas et al discloses a method for use by a first process executing in a computer system for interacting with a second process executing in the computer system, the method comprising:

a) upon execution of the first process, creating a copy of a global notification hook of the first process in the second process *(Col 7, Lines 16-63)*;

and

b) broadcasting a private startup message from the first process to the copy of the global notification hook *(Col 8, Lines 10-42)* wherein, in response to the private startup message, the copy of the global notification hook executes acts of:

i) determining whether subsequent messages passed between the operating system and a thread of the second process should be monitored *(Col 7, Lines 16-63)*; and

ii) in the event that subsequent messages should be monitored, activating executable code, wherein said executable code is configured to monitor the subsequent messages *(Col 7, Lines 16-63)*.

28. De Armas et al does not disclose said executable code consisting of a thread-level hook.

Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept messages directed to a window to modify the functionality of said window *(Teilhet; Chpt 18.2, Pgs 3-5)*.

29. It would have been obvious to one of ordinary skill in the art at the time of invention to modify the method disclosed by De Armas et al with Teilhet's teachings concerning the WH_CBT thread-level hook. One of ordinary skill would have been motivated by the fact that De Armas et al discloses the general use of thread-level hooks *(De Armas et al; Col 7, Lines 5-15)* and that Teilhet discloses that thread-level hooks can be used as a substitute for the functionality of the method of message interception disclosed by De Armas et al, subclassing *(Teilhet; Chpt 3.2, Pg 1, 4th paragraph)*.

30. **Claim 19:** De Armas et al disclose wherein the interception of messages is configured so as not to affect operation of a third process executing concurrently with the second process in the computer system *(Col 9, Lines 36-60) (Use of the "SendMessage()" API to send messages directly to the recipients of said message).*

31. **Claim 20:** De Armas et al discloses causing an action to occur in response to intercepted messages *(De Armas et al; Cols 11 and 12).*

32. **Claim 21:** De Armas et al and Teilhet disclose the method of **Claim 18**, wherein De Armas et al further discloses the action including creating a visual effect for a window of the second process *(Col 11, Line 65 - Col 12, Line 11).*

33. **Claim 22:** De Armas et al further discloses the global notification hook determines that subsequent messages should be monitored in the event that a window of interest exists in the second process *(Col 8, Lines 10-34) (A window of interest is one that currently has system focus or user input has been received from the user to indicate a window of interest).*

34. **Claim 23:** This claim is rejected for the same reasons as **Claim 22** above.

35. Claim 24: De Armas et al further discloses the first process is a desktop management process *(Col 5, Lines 26-49) (The TIS system intercepts messages to all applications and either modifies messages being transmitted or forwards them, to the necessary application*.

36. Claim 25: De Armas et al further discloses the second process is an application process *(Col 5, Lines 26-49)*.

37. **Claim 26:** De Armas et al further discloses the act of mapping executable code into an address space of the second process for the code that will intercept further messages *(Col 7, Lines 16 - Col 8, Line 9 and Col 9, Lines 18 - 35)*.

38. Claim 27: De Armas et al discloses a computer program product comprising:

a computer readable medium encoded with program code for a global notification hook of a first process, wherein the program code for the global notification hook is adapted to be copied into a second process during a startup sequence of the second process *(Col 7, Lines 16-63)(The installation of the global hook causes it to be installed upon startup of a new application)*, the program code for the global notification hook including:

a) program code for detecting an occurrence of a triggering message in the second process *(Col 8, Lines 5-34)(The "PostMessage()" API call is used to send the triggering message and the "GetMessage()" API call is used to retrieve said message)*;

b) program code for determining, in response to detecting the occurrence of the triggering message, whether subsequent messages passed between the operating system and a thread of the second process should be monitored*(Col 8, Lines 5-34).*

39. While De Armas et al discloses the interception of further messages and modifying the behavior of the window to which those messages are directed *(De Armas et al; Col 7, Lines 56-62)*, De Armas et al does not disclose program code for the use of a thread-level hook to intercept and modify said messages.

Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept messages directed to a window to modify the functionality of said window *(Teilhet; Chpt 18.2, Pgs 3-5).*

40. It would have been obvious to one of ordinary skill in the art at the time of invention to modify the computer program product disclosed by De Armas et al with Teilhet's teachings concerning the WH_CBT thread-level hook. One of ordinary skill would have been motivated by the fact that De Armas et al discloses the general use of thread-level hooks *(De Armas et al; Col 7, Lines 5-15)* and that Teilhet discloses that thread-level hooks can be used as a substitute for the functionality of the method of message interception disclosed by De Armas et al, subclassing *(Teilhet; Chpt 3.2, Pg 1, 4th paragraph).*

41. **Claim 28:** De Armas et al discloses causing an action to occur in response to intercepted messages *(Cols 11 and 12).*

42. **Claims 29-31:** De Armas et al and Teilhet disclose the method of **Claim 27**, wherein De Armas et al further discloses the program code is stored in a computer readable medium *(Col 13, Lines 12-26).*

43. While De Armas et al does not explicitly disclose the use of a magnetic storage medium, an optical storage medium, or a transmission medium as means for transporting data, it would have been obvious to one of ordinary skill in the art to use such means because said means are old and well-known in the art as the primary means for transporting computer program code.

44. Claims 32: De Armas et al discloses a computer program product comprising:

a computer readable medium encoded with program code for a global notification hook of a first process, wherein the program code for the global notification hook is adapted to be copied into a second process during a startup sequence of the first process, the program code for the global notification hook including:

a) program code for broadcasting a private startup message from the first process to the copy of the global notification hook *(De Armas et al; Col 7, Line 64 - Col 8, Line 34)*; and

b) program code for determining, in response to the private startup message, whether subsequent messages passed between the operating system and a thread of the second process should be monitored *(Col 8, Lines 5-34).*

While De Armas et al discloses causing an action to occur in response to intercepted messages *(Cols 11 and 12)*, De Armas et al does not disclose the computer program product being encoded with instructions for thread-level hooks to intercept said messages.

Teilhet discloses the use of the WH_CBT hook as a thread-level hook to intercept messages directed to a window to modify the functionality of said window *(Chpt 18.2, Pgs 3-5)*.

45. It would have been obvious to one of ordinary skill in the art at the time of invention to modify the computer program product disclosed by De Armas et al with Teilhet's teachings concerning the WH_CBT thread-level hook. One of ordinary skill would have been motivated by the fact that De Armas et al discloses the general use of thread-level hooks *(De Armas et al; Col 7, Lines 5-15)* and that Teilhet discloses that thread-level hooks can be used as a substitute for the functionality of the method of message interception disclosed by De Armas et al, subclassing *(Teilhet; Chpt 3.2, Pg 1, 4th paragraph)*.

46. **Claim 33:** De Armas et al discloses causing an action to occur in response to intercepted messages *(Cols 11 and 12)*.

47. **Claims 34-36:** De Armas et al further discloses the program code is stored in a computer readable medium *(Col 13, Lines 12-26)*.

48. While De Armas et al does not explicitly disclose the use of a magnetic storage medium, an optical storage medium, or a transmission medium as means for transporting data, it would have been obvious to one of ordinary skill in the art to use such means because said means are old and well-known in the art as the primary means for transporting computer program code.

Conclusion

49. The prior art made of record and not relied upon is considered pertinent to applicant's disclosure:

a) Clark et al (US Pat: 5,835,090) discloses a method and system for managing windows drawn on a desktop and the hooking of messages passed between the desktop manager and windows within the system;

b) Coffey et al (US PGPub: 2003/0040889) discloses a system and method for monitoring the usage of software on a computer by intercepting messages passed between the executing applications and the operating system; and

c) Lal (US PGPub: 2003/0048286) discloses a system and method for integrating the windows of different applications into one desktop window by intercepting and altering messages passed between the windows of said applications and the operating system.

50. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Richard Pantoliano Jr whose telephone number is (571) 270-1049. The examiner can normally be reached on Monday-Thursday, 8am - 4 pm EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, William Thomson can be reached on (571)272-3718. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

RP
10/30/06

MENG-AL T. AN
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2100